CLAIM AMENDMENTS

Claim Amendment Summary

Claims pending

- Before this Amendment: Claims 1-12, and 20-37
- After this Amendment: Claims 1-12, and 20-37

Non-Elected, Canceled, or Withdrawn claims: None

Amended claims: 1, 12, 20, and 26

New claims: None

Claims:

1. (Currently Amended) A computer-implemented method for processing data, the method comprising:

executing a pipeline of commands (pipeline), wherein a pipeline is a plurality of commands entered as a single command string on a command line, each particular command separated from each other particular command by a delimiter and executed serially;

receiving a parseable object emitted from a prior object-based command within ~~a~~ the pipeline comprising a plurality of object-based commands, the prior object-based command being one of the plurality of object-based commands, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with the prior object-

based command within the pipeline through the parseable object emitted from the prior object-based command, wherein the parseable object includes at least one method, and wherein an operating environment that supports the pipeline of the plurality of object-based commands is configured to support execution of the object-based commands within the same process;

obtaining a data type for the parseable object;

obtaining format information describing a format for the data type; and

emitting a format object for access by another subsequent object-based command, the format object being based on the format information wherein the format object is emitted to a computer readable storage medium.

2. (Original) The computer-implemented method of claim 1, wherein obtaining format information comprises accessing an XML-based document.

3. (Previously Presented) The computer-implemented method of claim 1, wherein the subsequent object-based command comprises an output command configured to render results of the pipeline based on the received parseable object and the format object.

4. (Original) The computer-implemented method of claim 3, wherein the rendering of the results comprises displaying on a console.

5. (Original) The computer-implemented method of claim 3, wherein the rendering of the results comprises importing the results into an application.

6. (Original) The computer-implemented method of claim 3, wherein the rendering of the results comprises displaying in a graphical user interface.

7. (Previously Presented) The computer-implemented method of claim 1, wherein the other subsequent object-based command comprises a markup command configured to add property annotation to selected parameters within the parseable object and emitting these property annotations for input by further subsequent object-based commands in the pipeline.

8. (Previously Presented) The computer-implemented method of claim 1, wherein the other subsequent object-based command comprises a convert command configured to convert the received parseable object into a specific format.

9. (Original) The computer-implemented method of claim 8, wherein the specific format comprises an XML document, an Active Directory Object, or a comma separated value format.

10. (Previously Presented) The computer-implemented method of claim 8, wherein another subsequent object-based command comprises a transform command that receives the specific format from the convert command and transforms the specific format into another specific format based on a style sheet.

11. (Original) The computer-implemented method of claim 1, wherein the format information describes the data type and at least one of a shape, a property, or a header.

12. (Currently Amended) A computer readable storage medium including at least one tangible component and having computer-executable instructions that, when executed, direct a computing system to perform a method for providing pre-output processing and data based upon input from a prior command's output data, the ~~instructions~~ method comprising:

executing a pipeline of object-based commands (pipeline), wherein a pipeline is a plurality of commands entered as a single command string on a command line, each particular command separated from each other particular command by a delimiter and executed serially;

receiving by reference a parseable pipeline object (PPO) from a computer readable storage medium, the PPO having been emitted from a prior object-based command within an administrative tool framework that supports ~~a~~ the pipeline ~~of a plurality of object-based commands~~ and, the administrative tool framework is configured to support the execution of the object-based commands within the same computer process, the prior object-based command being one of the plurality of ~~object-based~~ commands, wherein the receiving occurs as part of the pipeline of object-based commands entered together as a parseable stream and separated into separate commands, such that a subsequent object-based command within the pipeline which receives the parseable object is

configured to communicate with the prior object-based command within the pipeline through the parseable object emitted from the prior object-based command, the parseable object having at least one method;

obtaining a data type for the PPO using object reflection;

obtaining format information describing a format for the data type of the PPO, wherein the format information describes at least one of a plurality of formats, the plurality of formats comprising:

a shape;

a property; and

~~a header; and~~ a header,

wherein the format information is obtainable by accessing one of a plurality of data sources, wherein the data source is one selected from the group consisting of: an XML document, an Active Directory Object, and a delimiter separated values file; ~~and~~

emitting to a computer readable storage medium an output format object (OFO) for access by another subsequent object-based command from the plurality of object-based commands, wherein the OFO is based upon the obtained format information, and parameters of the command; and

terminating the pipeline is an output command that accepts as input the PPO and the OFO and delivers the result of the pipeline of object-based ~~commands:~~ commands,

~~wherein~~ wherein:

results are delivered to an output method that has been provided by the administrative tool framework to support the methods of output supported by the computer; and

~~wherein~~ the format of result depends upon whether the output command ~~command~~ is preceded by any number of format modifying commands such that:

in an event that a format modifying command includes a markup command, the format modifying command will add property annotation to selected parameters within the PPO for input by further subsequent commands in the pipeline; ~~and~~

in an event that a format modifying command includes a convert command, the format modifying command will be configured to convert the PPO into a specific file format; and

in an event that a format modifying command includes a transform command, the format modifying command will be configured to receive instruction from a format modifying command including a convert command and transform the PPO from the specific file format into another specific format based upon a style sheet.

13-19. (Canceled)

20. (Currently Amended) A system that supports data driven output, the system comprising:

a processor;

a memory, the memory being allocated for a plurality of computer-executable instructions which are loaded into the memory for execution by the processor, wherein upon execution of the computer-executable instructions the system being configured to:

execute a pipeline of object-based commands (pipeline), wherein a pipeline is a plurality of commands entered as a single command string on a command line, each particular command separated from each other particular command by a delimiter and executed serially;

receive a parseable object emitted from a prior object-based command within an operating environment that supports ~~a~~ the pipeline ~~of a plurality of object-based commands~~ and that is configured to support the execution of the object-based commands within the same process, the prior object-based command being one of the plurality of object-based commands, wherein receiving the parseable object occurs as part of the pipeline, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with the prior object-based command within the pipeline through the parseable object emitted from the prior object-based command, the parseable object having at least one method;

obtain a data type for the parseable object;

obtain format information describing a format for the data type; and

emit a format object for access by a subsequent object-based command from the plurality of object-based commands, the format object being based on the format information wherein the format object is emitted to a computer readable storage medium.

21. (Previously Presented) The system of claim 20, wherein the format information comprises accessing an XML-based document.

22. (Previously Presented) The system of claim 20, wherein the format information describes the data type and at least one of a shape, a property, or a header.

23. (Previously Presented) The system of claim 20, wherein the other subsequent object-based command comprises a markup command configured to add property annotation to selected parameters within the parseable object and emitting these property annotations for input by further subsequent object-based commands in the pipeline.

24. (Previously Presented) The system of claim 20, wherein the other subsequent object-based command comprises a convert command configured to convert the received parseable stream into a specific format.

25. (Previously Presented) The system of claim 20, wherein another subsequent object-based command comprises a transform command that

receives the specific format from the convert command and transforms the specific format into another specific format based on a style sheet.

26. (Currently Amended) A method for providing a data driven command line output, the method comprising:

receiving a command-line instruction containing an output command configured to receive a parseable object, the parseable object having at least one method, wherein the receiving occurs as part of a pipeline of a plurality of object-based commands, wherein a pipeline is a plurality of commands entered as a single command string on a command line, each particular command separated from each other particular command by a delimiter and executed serially, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with a prior object-based command within the pipeline through the parseable object emitted from the prior object-based command;-and

executing the output command to manipulate the parseable object and to output a result to an output destination.

27. (Previously Presented) The method of claim 26, wherein the command line instruction is received and the output command is executed in an object-based command-line environment.

28. (Previously Presented) The method of claim 27, wherein the output command is provided by the command-line environment.

29. (Previously Presented) The method of claim 26, wherein outputting the result comprises displaying the results on a console.

30. (Previously Presented) The method of claim 26, wherein outputting the result comprises importing the results into an application.

31. (Previously Presented) The method of claim 26, wherein outputting the result comprises displaying the results in a graphical user interface.

32. (Previously Presented) The method of claim 26, further comprising another command configured to provide the object to the output command.

33. (Previously Presented) The method of claim 32, wherein the other command comprises a format command configured to emit display information associated with the object.

34. (Previously Presented) The method of claim 33, wherein the output command ignores the display information when outputting the result.

35. (Previously Presented) The method of claim 34, wherein the other command comprises a markup command configured to add a property annotation to a parameter within the object.

36. (Previously Presented) The method of claim 32, wherein the other command comprises a convert command configured to convert the object into a specific format.

37. (Previously Presented) The method of claim 36, wherein the specific format comprises an XML document, an Active Directory Object, or a comma separated value format.